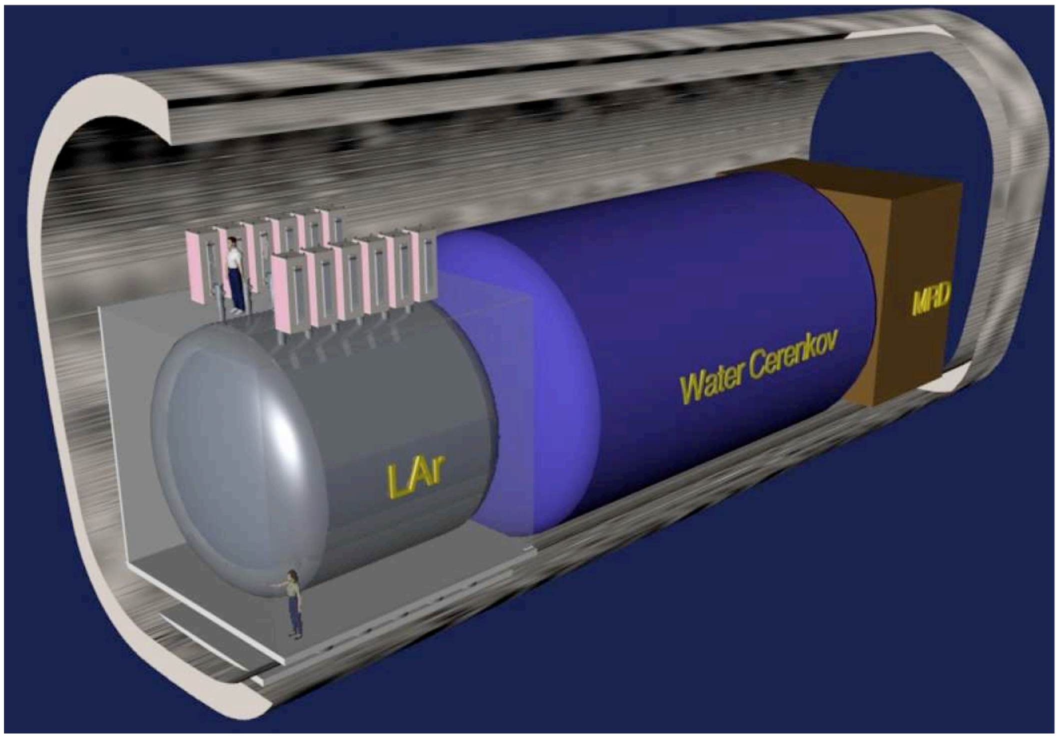
Geant4 based Water Cherenkov detector simulation

C. Walter / Duke University

The T2K 2KM MC



For T2K we designed a complex 2KM away from the neutrino source which included a 1 kton WC detector optimized to have the same performance as SK.

We used our experience from K2K and didn't try to copy the orientation and tube size of SK.

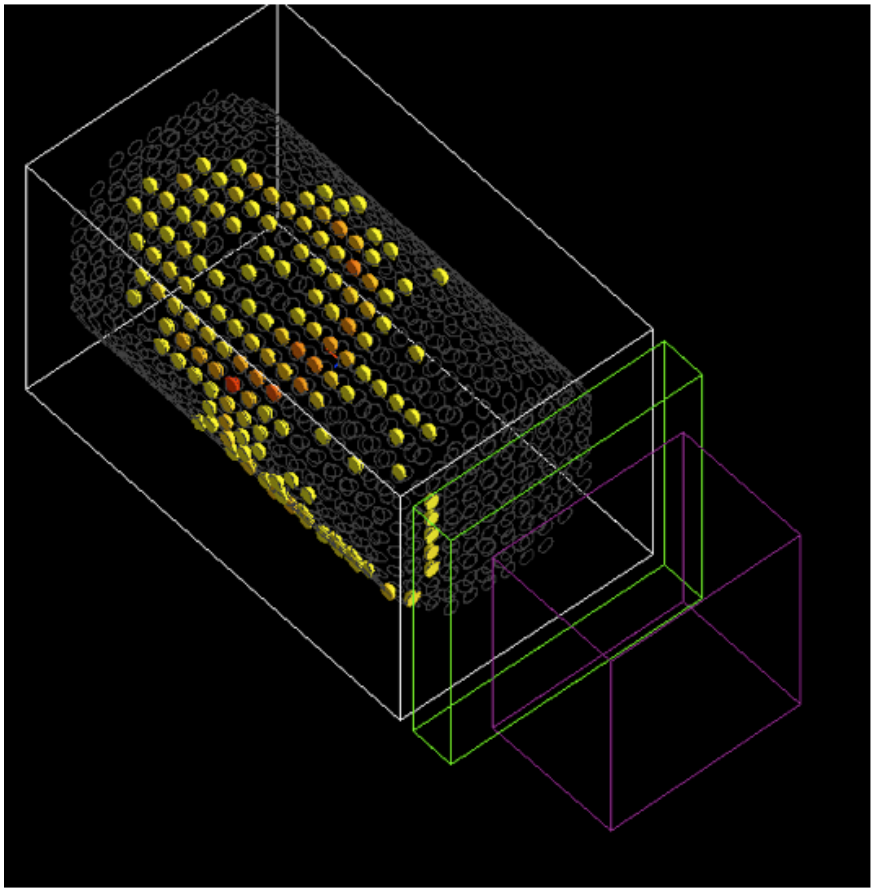
We built a flexible system in Geant4 and used it to simulate several geometries including the K2K tank so we could tune the simulation with real data.

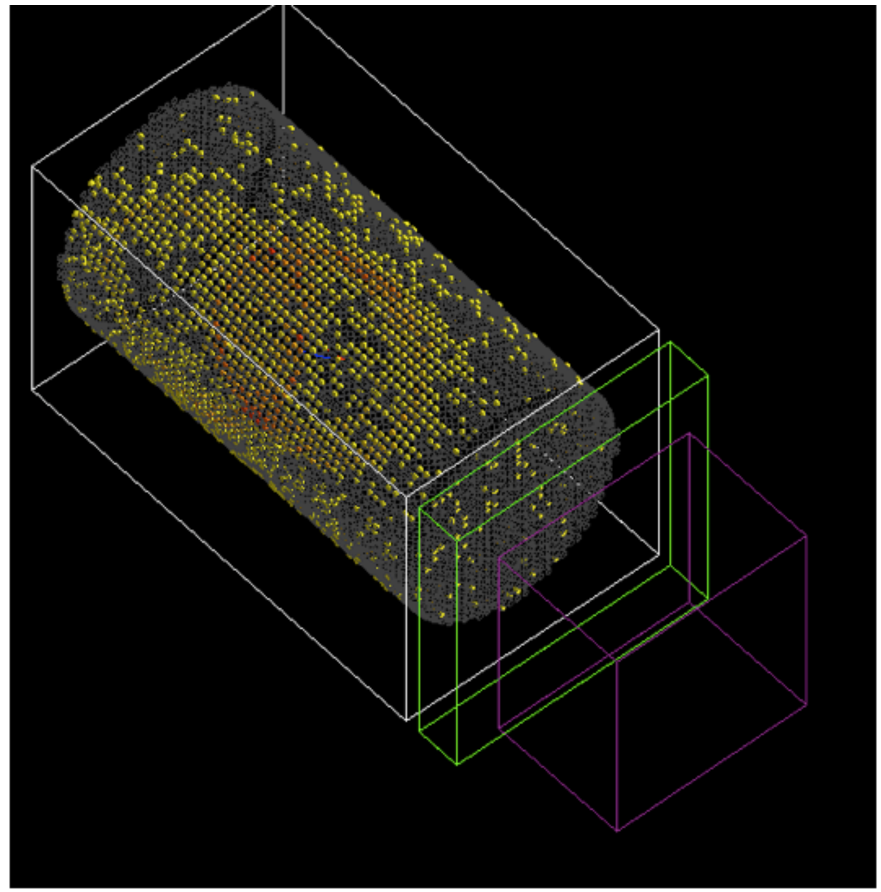
I am using this code as a base to build a new DUSEL WC simulation.

2KM MC Characteristics and Features

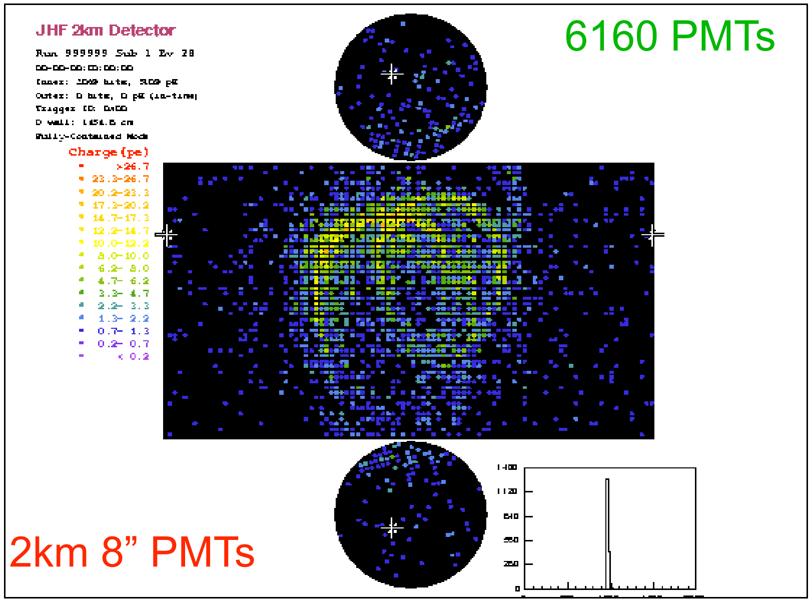
- WC + Lar or FGD + MRD simulation
- Both 1kton and 2KM WC configurations with 8” and 20” PMTs
  - Tube placement handled programmatically.
- On the fly configuration via messengers
  - Geometry setup
  - Hadronic models
  - Generator type
  - Input/output file names
- Can read NUANCE format vector files (also have NEUT-> NUANCE converter)
- Root and PAW based outputs
- Raw and digitized hits based on SK-III style electronics
- Simplified pmt structure
- Water and reflective material parameters tuned to SK and K2K 1Kton data.
- Specialized G4 extensions to attach information to each photon
- Outputs geometry information for use by downstream programs
- External root based event display
- External conversions to SK format for reduction, display and and reconstruction

Variable PMT size

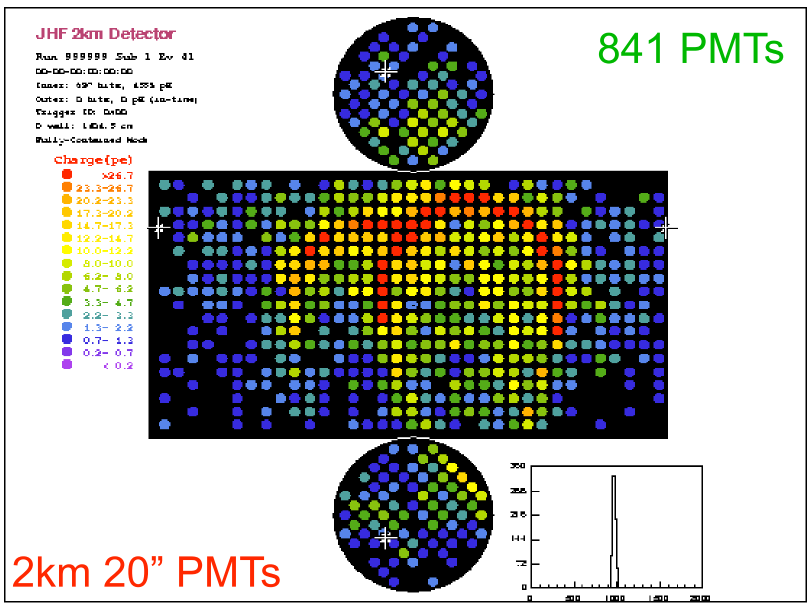


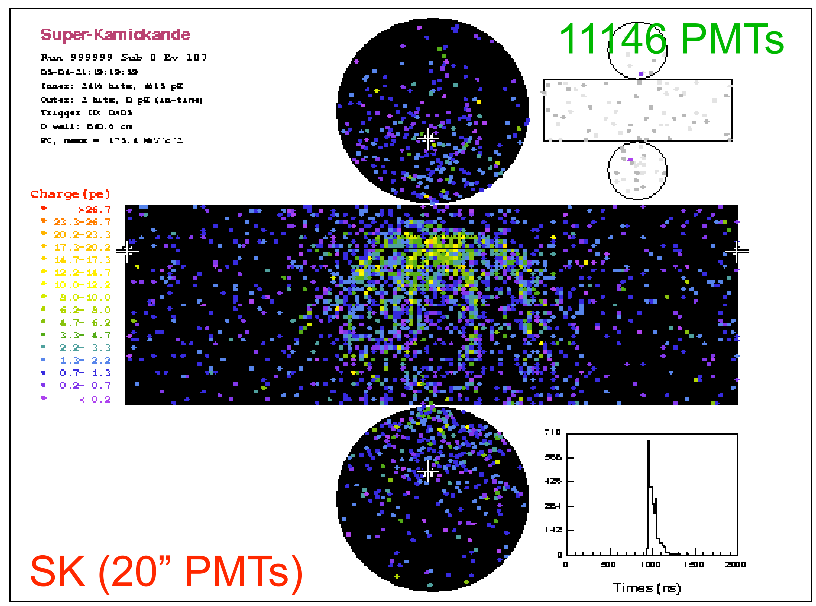


Comparison of pizero in GEANT4 8” and 20” 2km WC and SK viewed with SK event display.



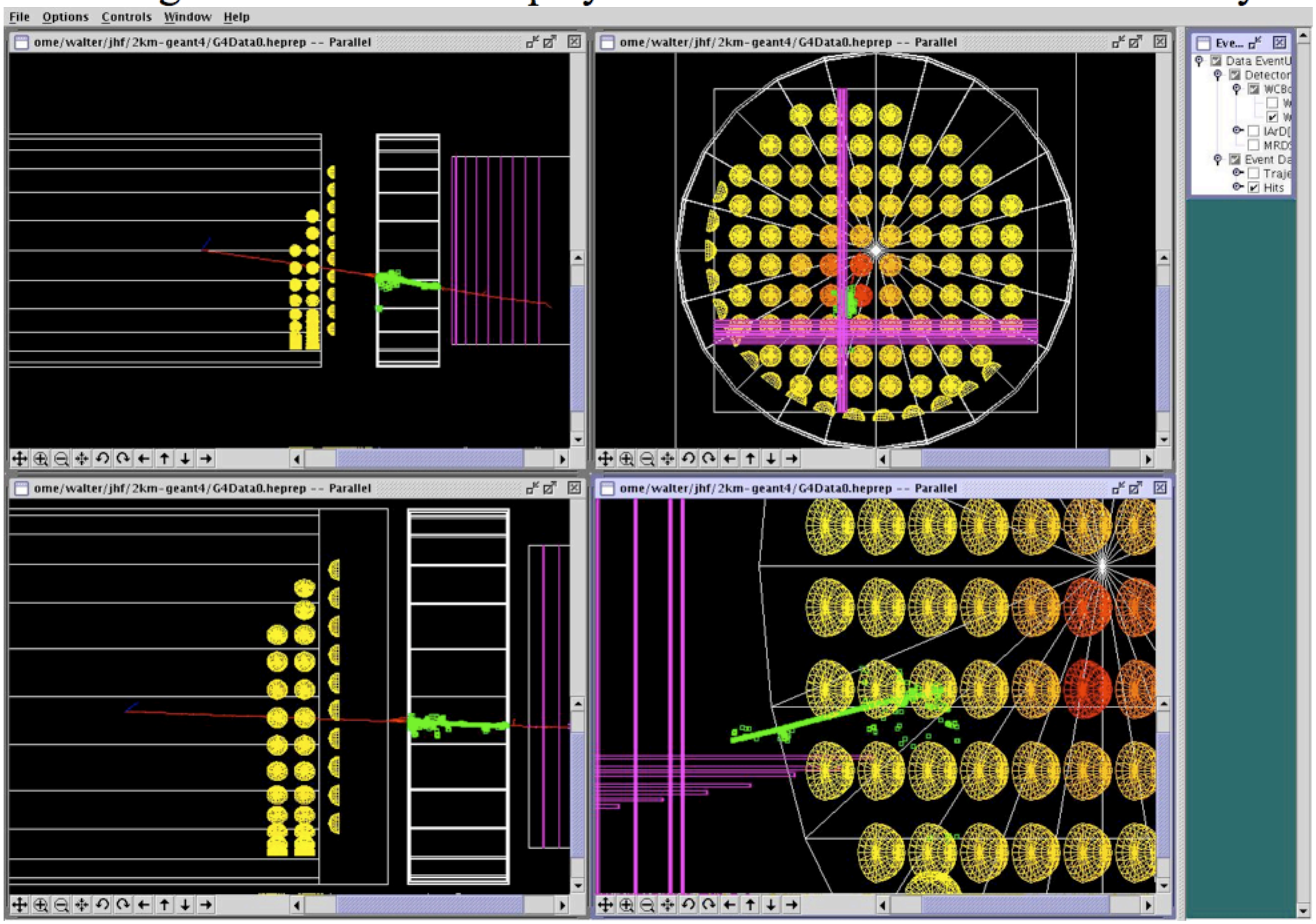
Also simulate 1kton at K2K
analyze data with modified SK tools and
compare with real neutrino data



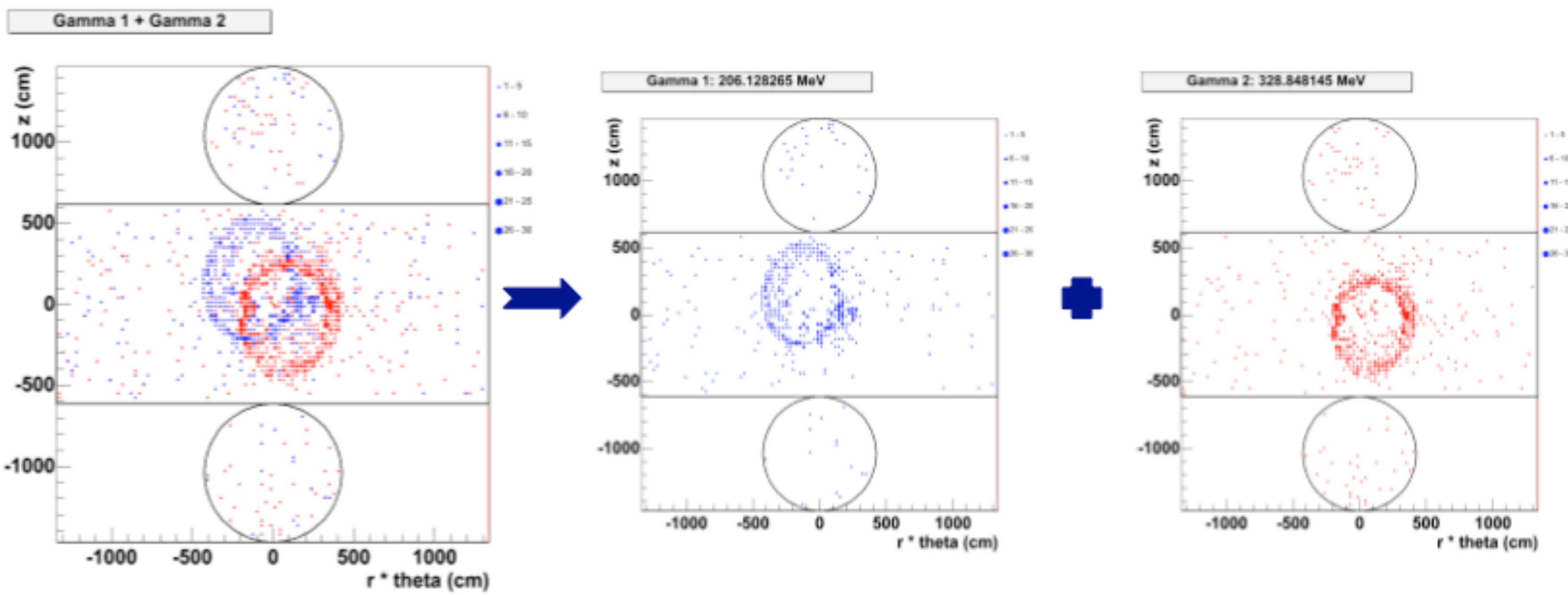


Multiple detectors and configurations

WIRED generalized event display that can read HEPREP files made by G4

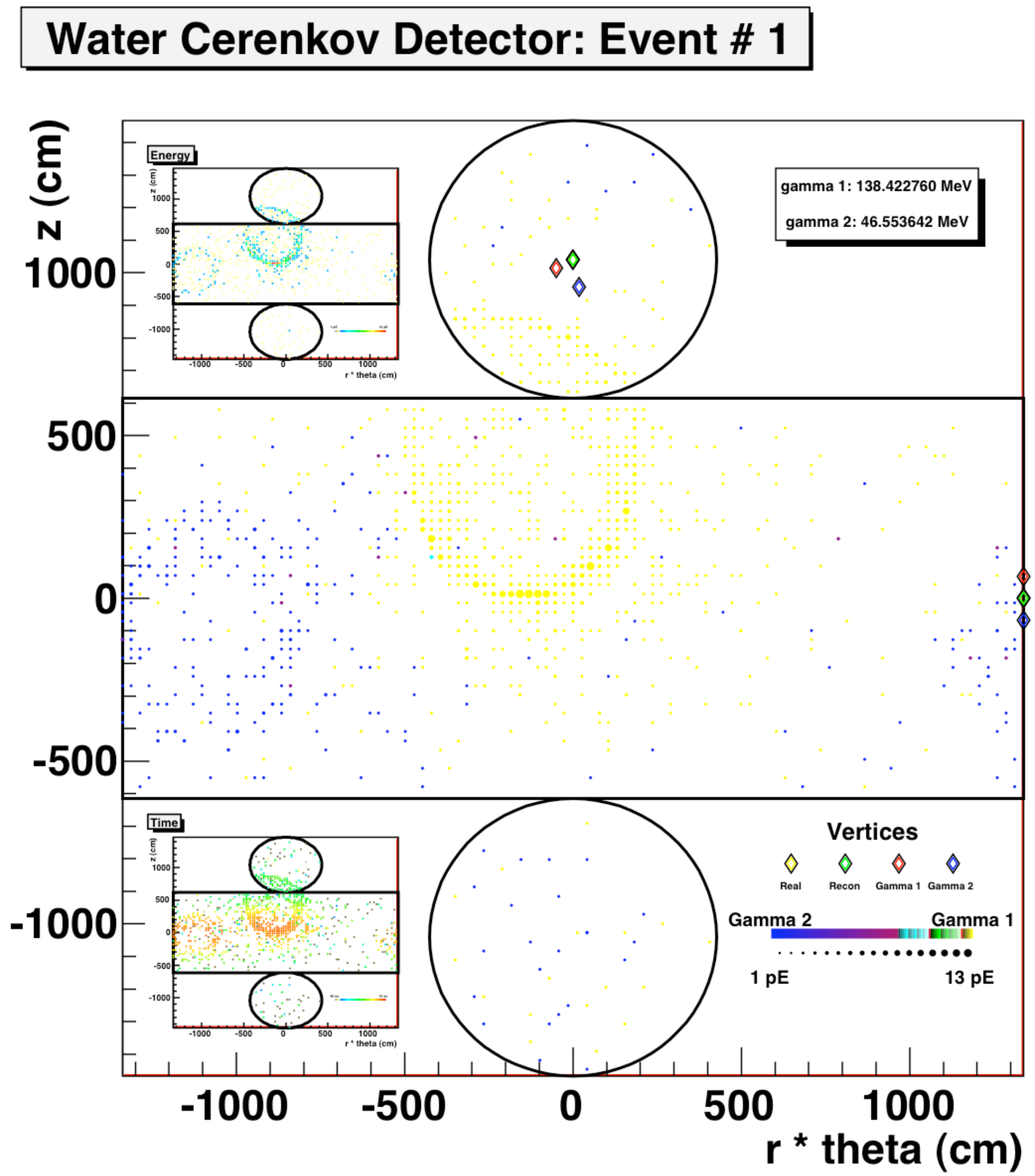


Raw ROOT based event display with specialized features for pizero studies.



Same Pi0 event separated by parent γ in Root display (γ_1 = blue, γ_2 = red)

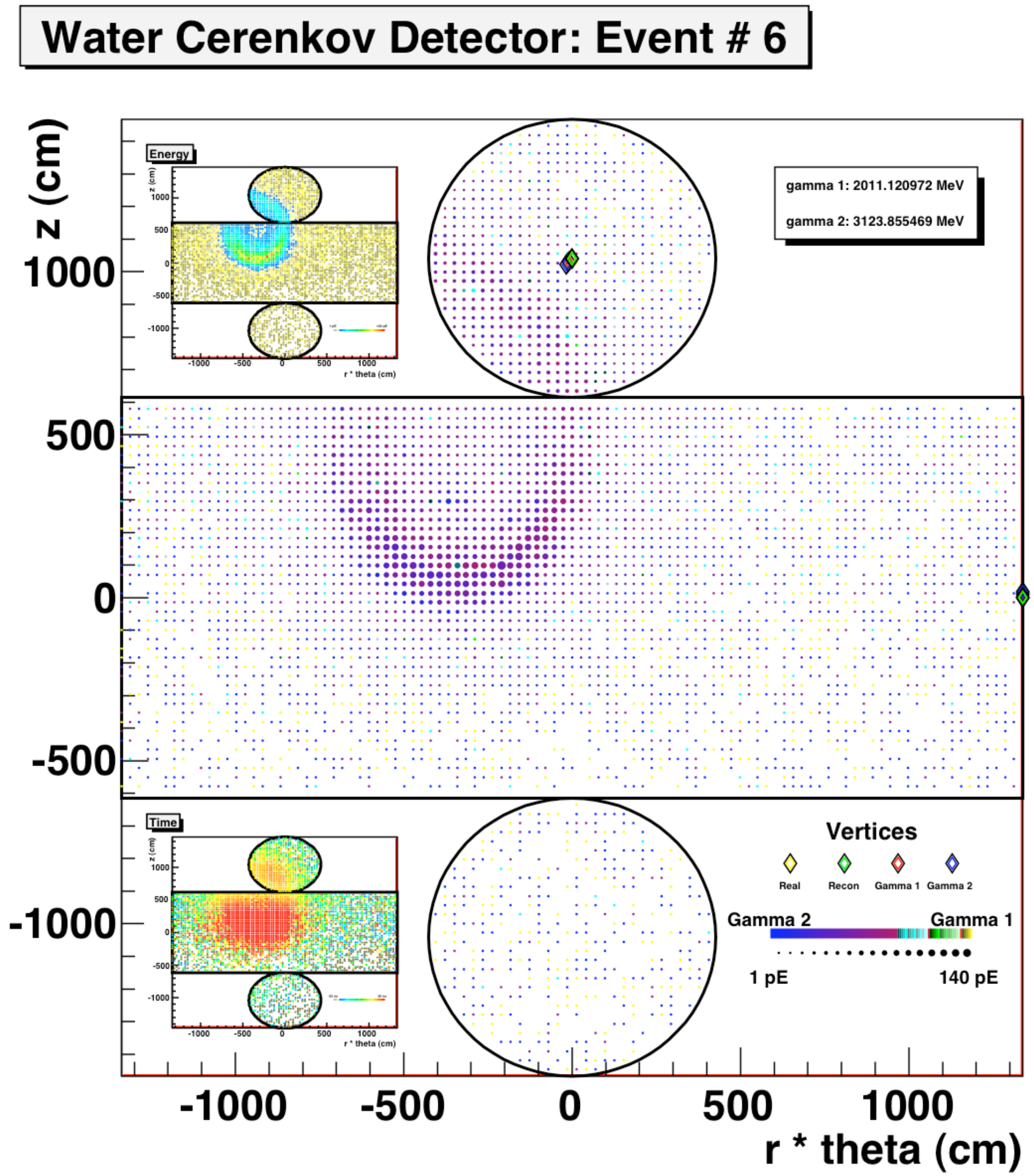
Modified: WC and Root only



Example of visualization of two gammas decayed from a Pi0

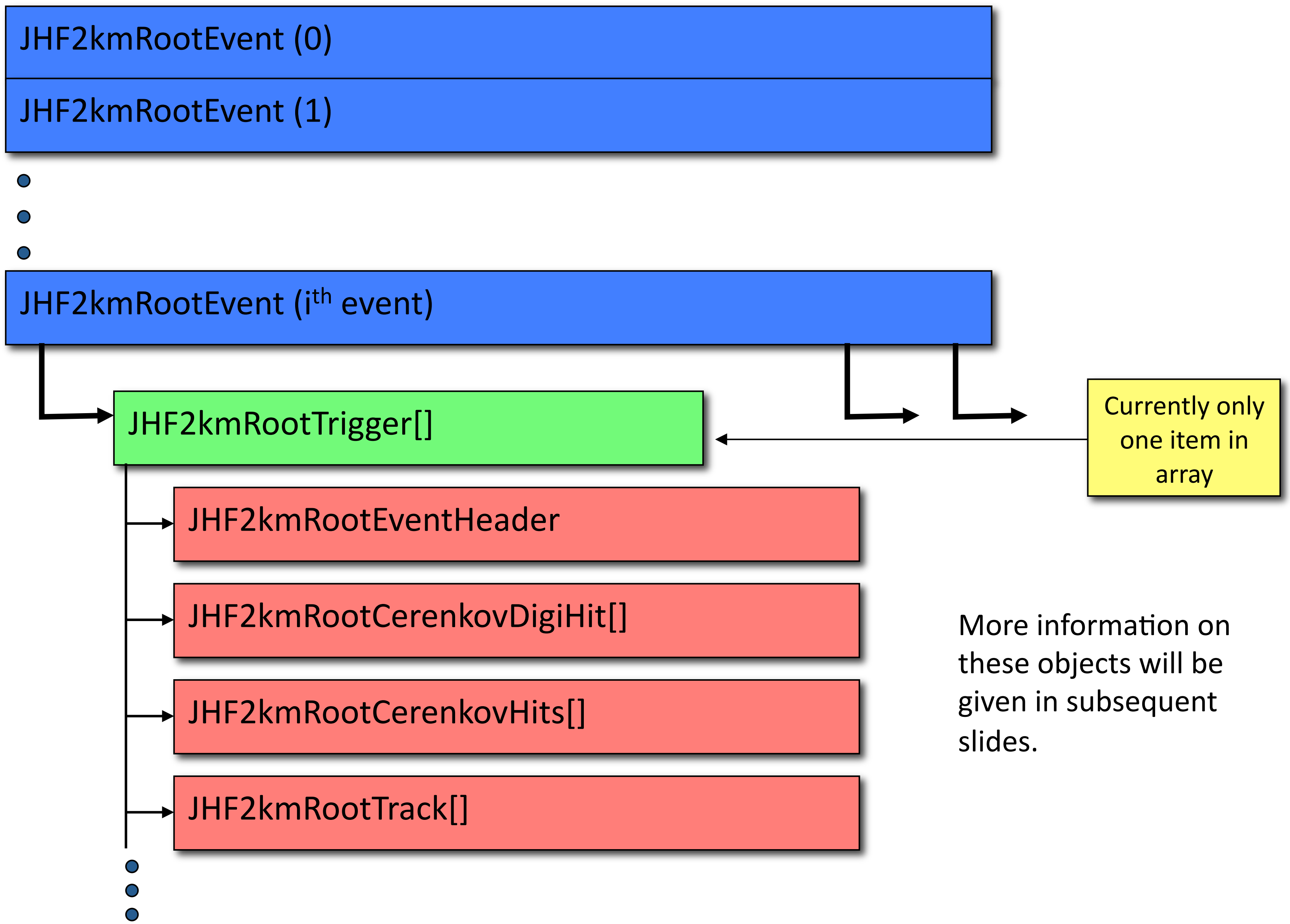
2 GeV muon run through the 2KM detector.

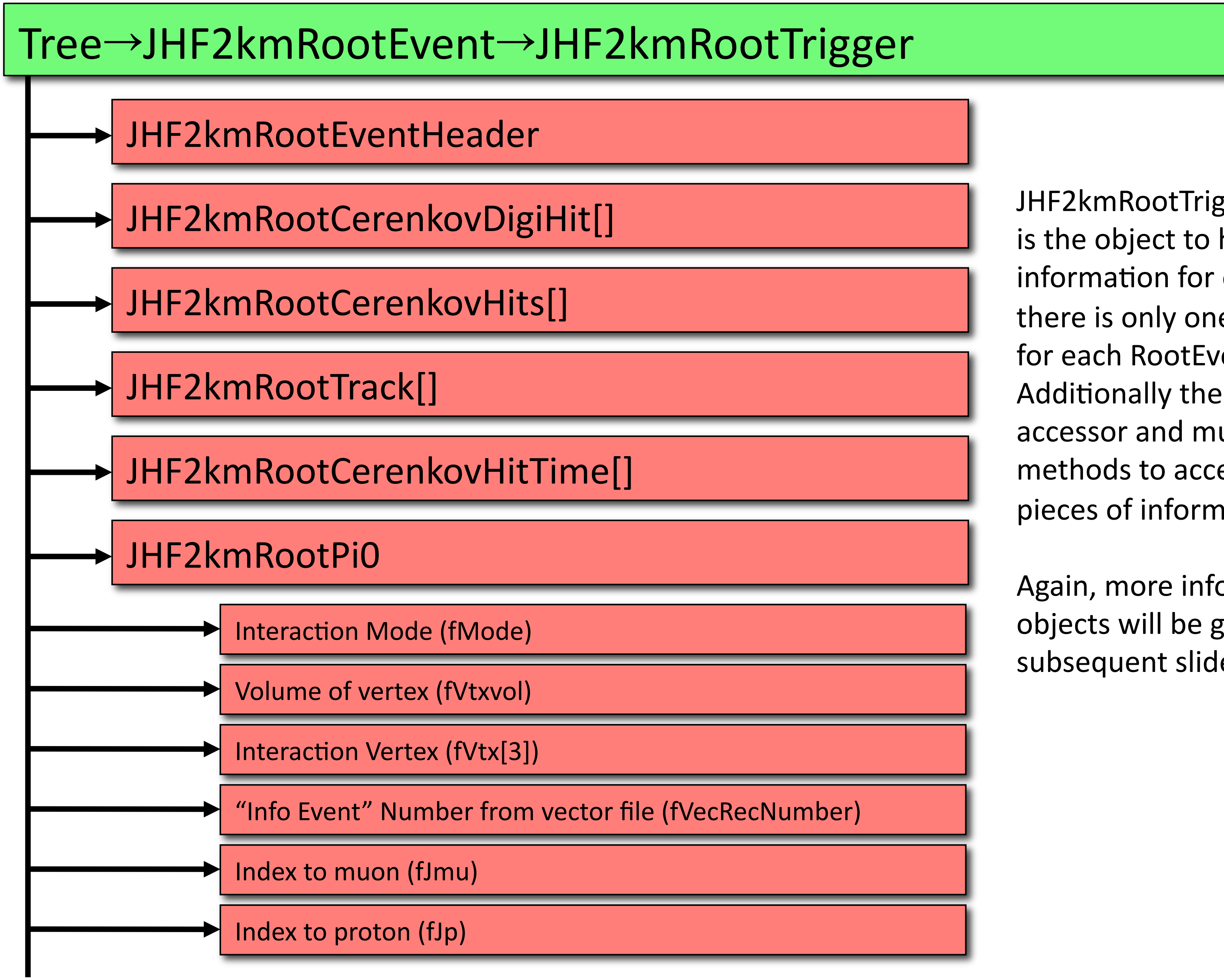
~ 10 sec/event



Reconstructed visualization from ROOT file information

Recorded Root information



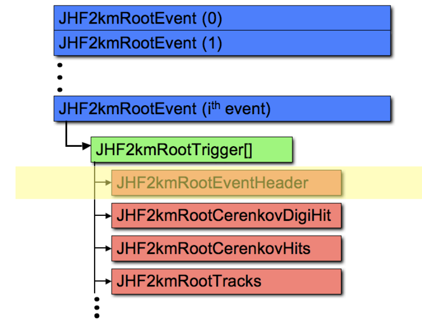


JHF2kmRootTrigger generally is the object to hold all of the information for each event (as there is only one RootTrigger for each RootEvent). Additionally there are accessor and mutator methods to access all of these pieces of information.

Again, more information on objects will be given in subsequent slides.

Tree→JHF2kmRootTrigger→JHF2kmRootEventHeader

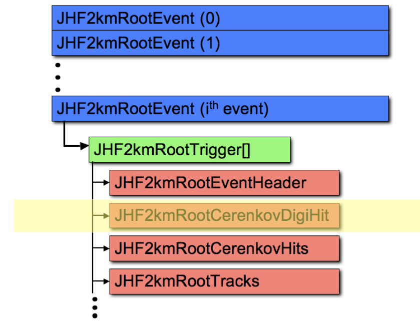
- Event Number (Int_t fEvtNum)
- Date (Int_t fDate)
- Subevent Number (Int_t fSubEvtNum)
- Run Number (Int_t fRun)



The event header stores general information about the event. Some of these values (such as the subevent number) may just be set to zero.

Tree→JHF2kmRootTrigger→JHF2kmRootCerenkovDigiHit

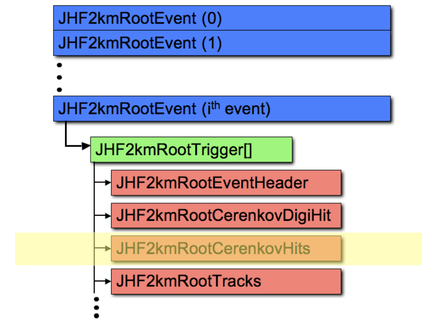
- Tube ID (Int_t fTubeId)
- Charge (Int_t fQ)
- Time (Int_t fT)



Each object in this array contains the basic information for a PMT hit.

Tree→JHF2kmRootTrigger→JHF2kmRootCerenkovHit

- Tube ID (Int_t fTubeId)
- fTotalPe[2]
  - fTotalPe[0] is starting index in CerenkovHitTime array
  - fTotalPe[1] is number of objects in CerenkovHitTime array associated with this CerenkovHit

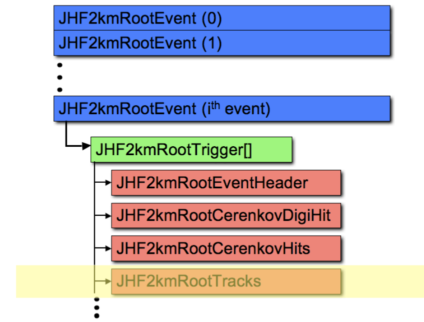


JHF2kmCerenkovHitTime will be explained in more detail later.

This array and the JHF2kmCerenkovHitTime array will only be filled if the system variable _SAVE_RAW_HITS is defined. This object will contain (in fTotalPe[1]) the total number of photoelectrons that hit the PMT. The value fTotalPe[0] can be used to look in the JHF2kmCerenkovHitTime array to get more details about the individual photoelectron hits.

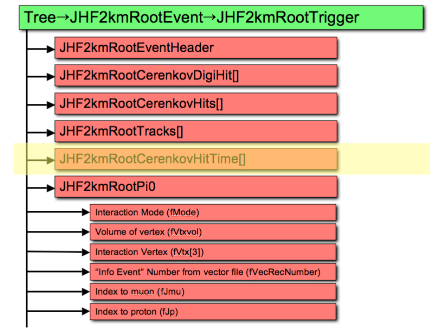
Tree→JHF2kmRootTrigger→JHF2kmRootTrack

- ID of final state particle (fIpnu)
- Flag (fFlag): -1 is incoming neutrino, -2 target, 1 outgoing lepton, 2 most energetic outgoing nuclean
- Mass of final state particle (fM)
- Momentum of final state particle (fP)
- Energy of final state particle (fE)
- Starting volume of final state particle (fStartvol)
- Stopping volume of final state particle (fStopvol)
- Direction of final state particle (fDir[3])
- Momentum-vector of final state particle (fPDir[3])
- Stopping point of final state particle (fStop[3])
- Starting point of final state particle (fStart[3])
- ID of parent particle (fParenttype) ??????IS THIS RIGHT????
- Time of particle creation (fTime)



Tree→JHF2kmRootTrigger→JHF2kmRootCerenkovHitTime

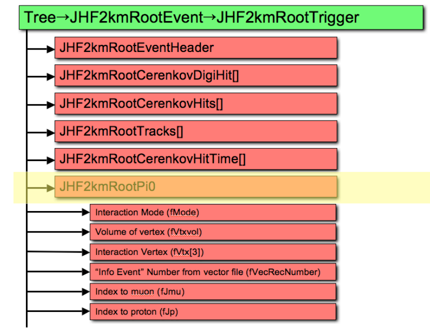
- True time (fTruetime)
- Parent ID (fPrimaryParentID)



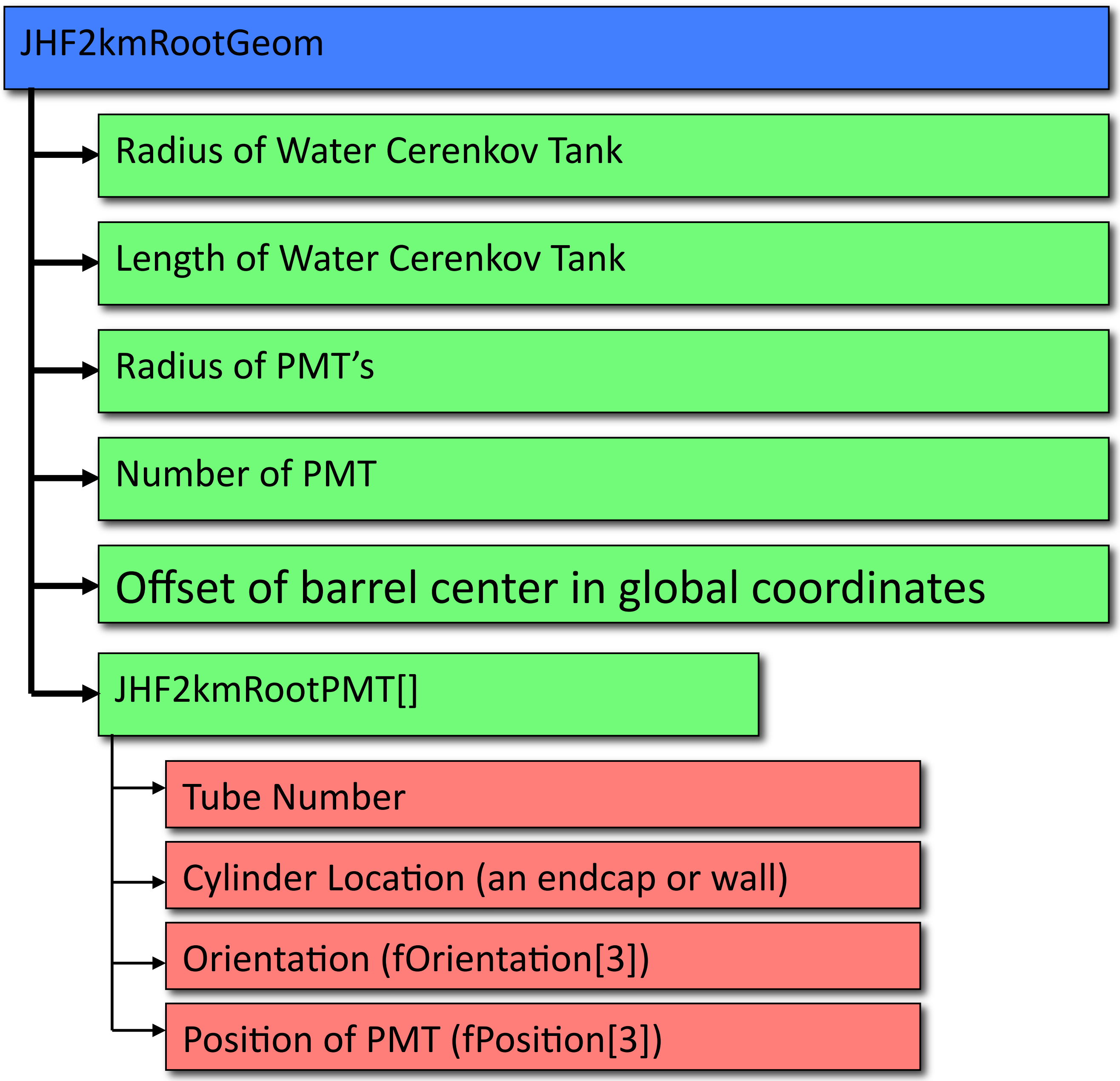
As previously described, this is a large array of all raw photoelectron hits. The parent ID can also be used to see specifically where the photoelectron came from. In the visualization software, we use this value to display different colors for the two different decay gammas from a Pi0.

Tree→JHF2kmRootTrigger→JHF2kmRootPi0

- Pi0 starting point (fPi0Vtx)
- IDs of decay gammas (fGammaID[2])
- Energy of decay gammas (fGammaE[2])
- Starting point of decay gammas (fGammaVtx[2][3])



In the case of the particle being a Pi0, we save this information to aid with reconstructing the resulting cerenkov light from the two decay gammas.



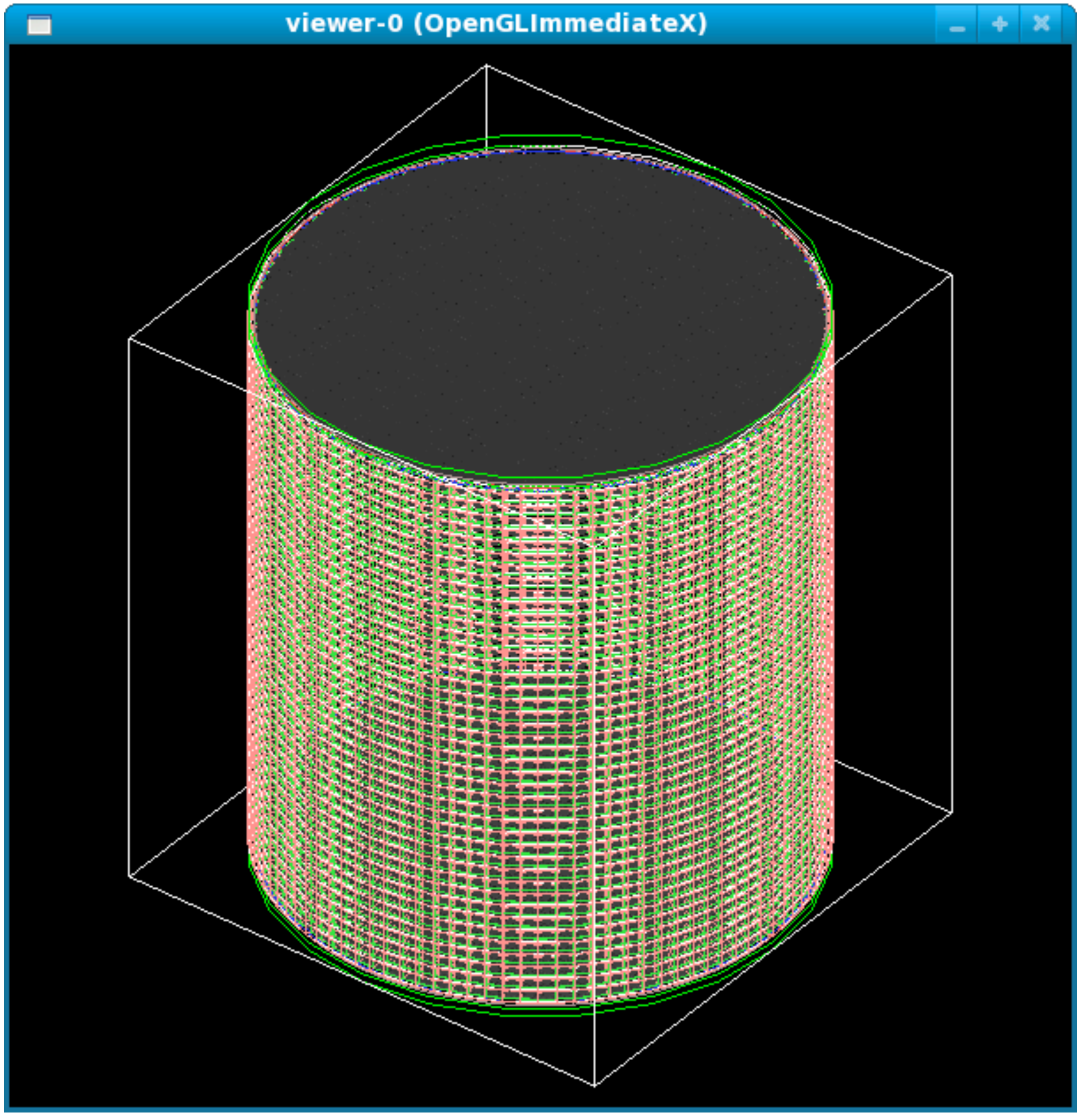
The RootGeom is stored in a separate ROOT file. The structure specifies the size of the detector and the layout of the PMT's.

Modifications under way

- Converting to Water Cherenkov only.
- Root output only
- Addition of SK size and larger DUSEL size tank options and orientations.
- Removal of specific T2K/SK code.
- Setup for non-programmatic tube placement.

Crude Super-K III configuration

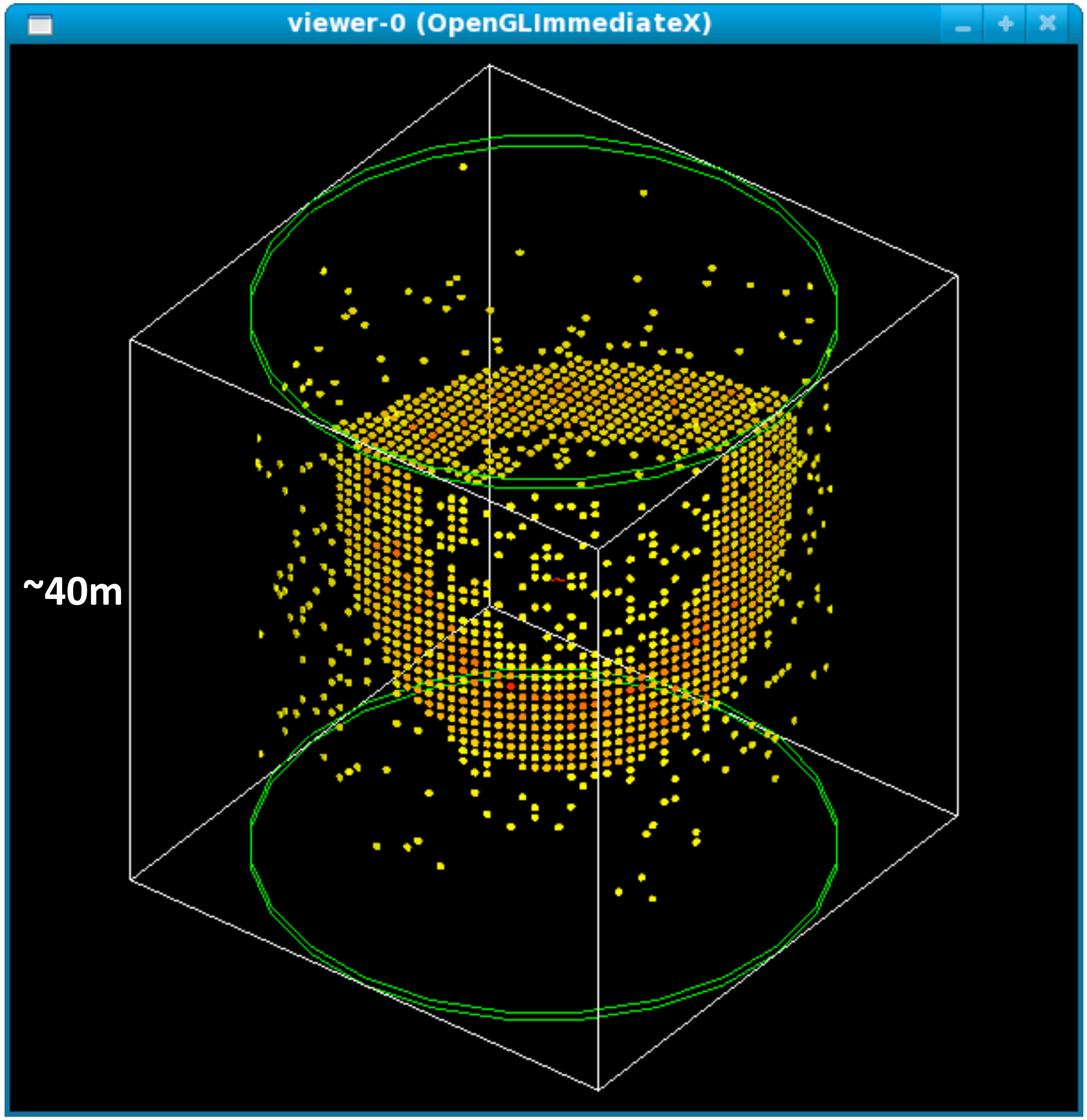
Missing exact tube placement and detailed PMT and FRP model.



WC Detector in debug mode, showing all PMT cells and black sheet.

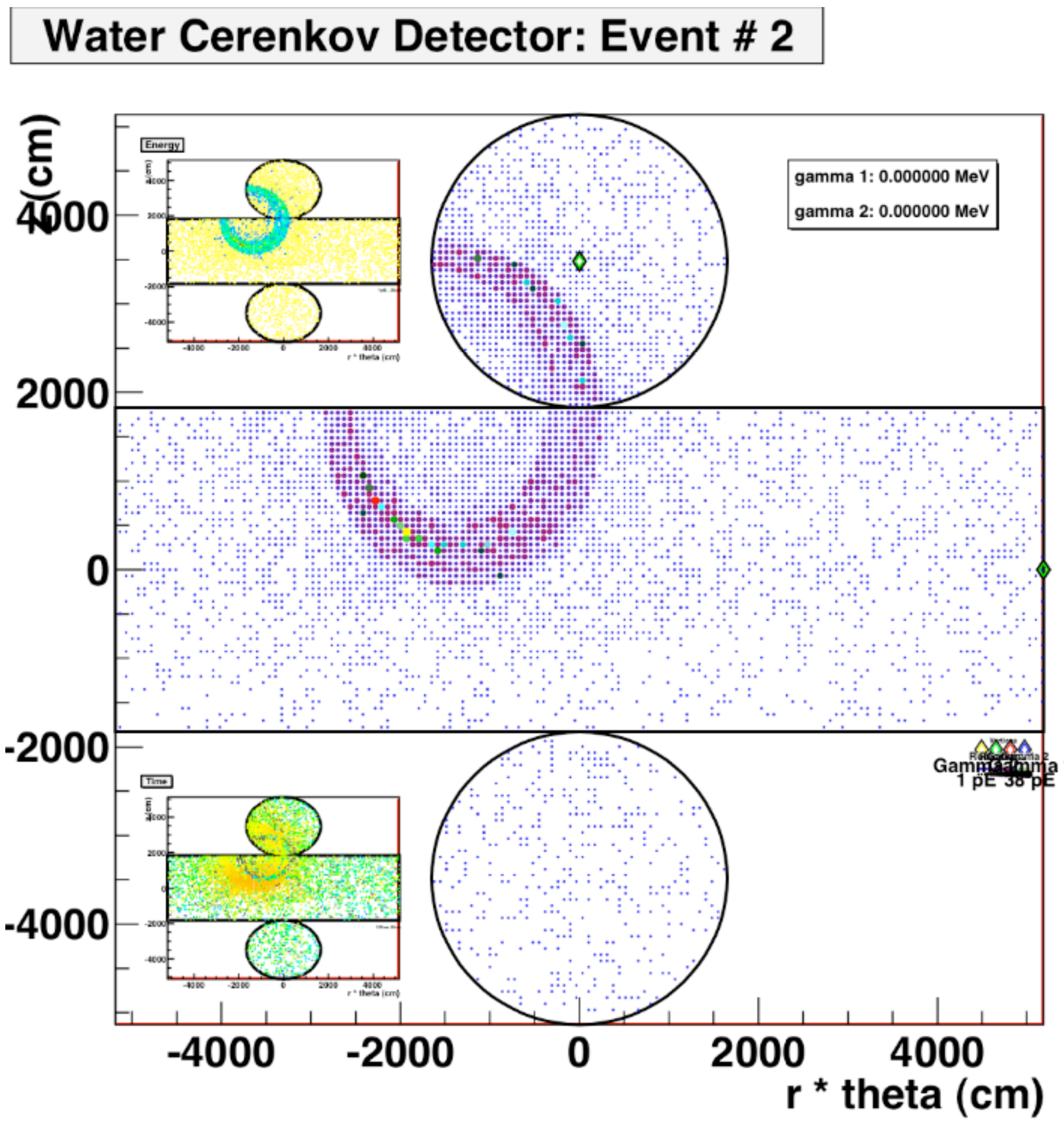
2 GeV muon run through the detector.

~ 15 sec/event



WC Detector scaled to approximately Super Kamiokande size

Root based evt display for SK

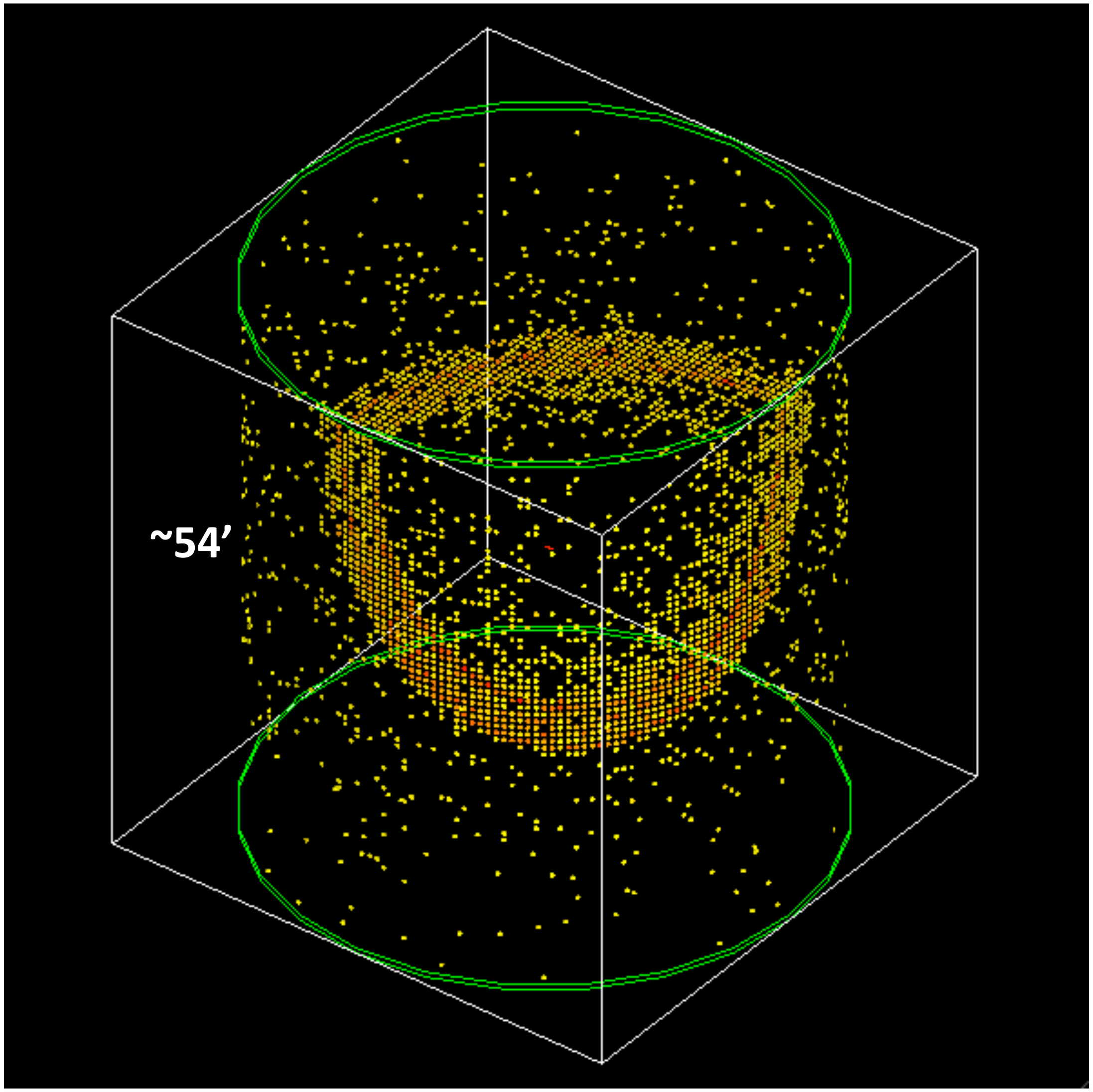


DUSEL Size (53' x 54')

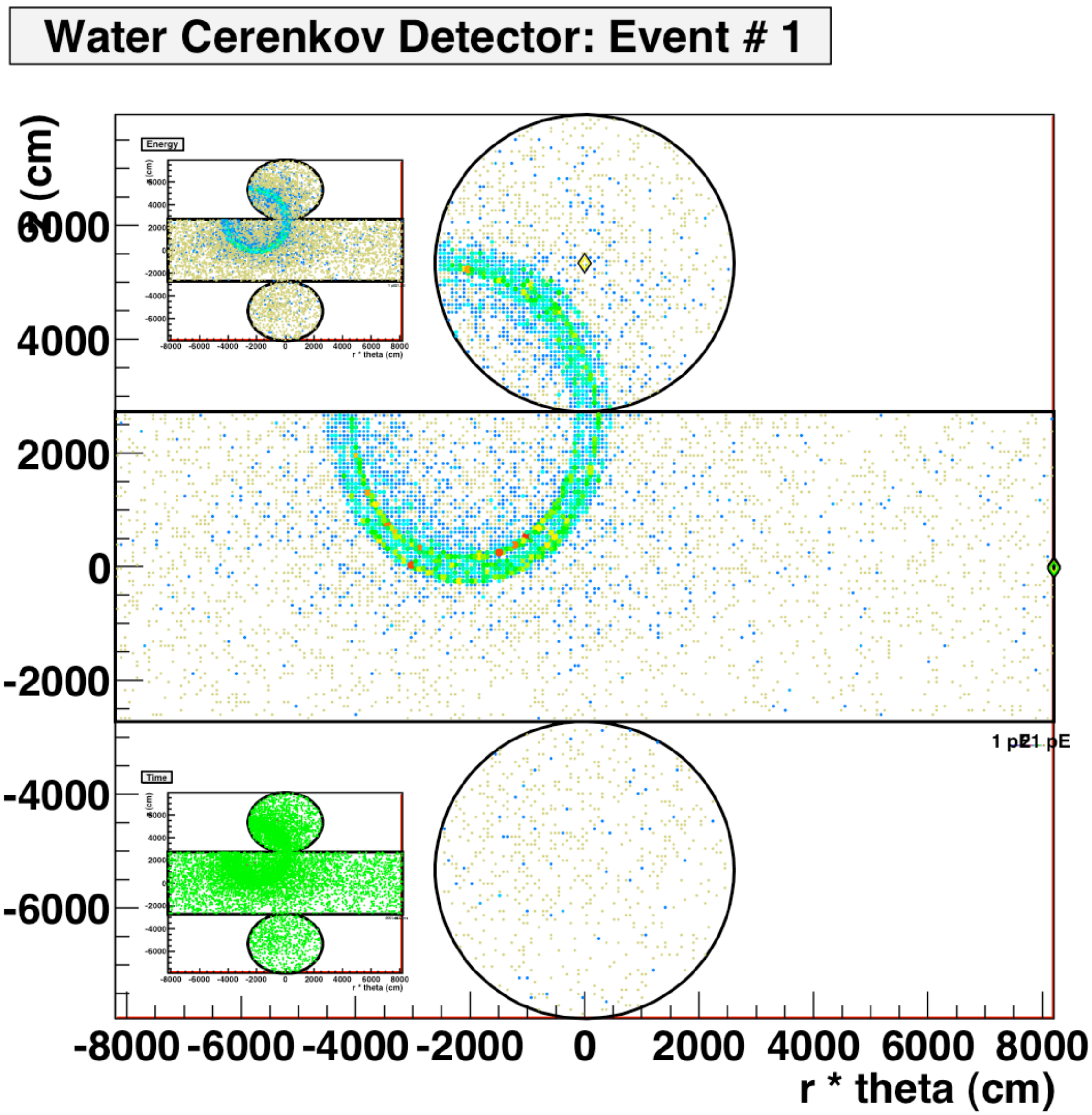
2 GeV muon run through
the detector.

26720 20" tubes used for this
configuration

~ 30 sec/event



Root Display



Next immediate steps

- Modify root display tools to handle bigger geometry and different event classes.
- Finish generalizing the code.
- Determine how to do tube placement
  - Important for SK validation
  - Exploring GDML which seems to be a the standard way of handling this in G4 and ROOT now.
- GDML is an XML based text description of materials and geometries.

Can be read in and out of G4 and ROOT and other solids programs.

Should allow for tube placement etc without recompilation.

XML Declaration

GDML Namespace

XML Schema Instance Namespace

GDML Schema Location

Declaration of constants, positions, rotations, scalings, variables and matrices

Definition of elements, materials, isotopes etc. *Only materials can be referenced to volumes*

Simple material defined using fractions of elements defined higher up.

All solids to be used inside the geometry defined here

All volumes declared in the structure section.

World volume is declared

Physical Volumes placed inside a Logical Volume with a specified position, rotation and scale

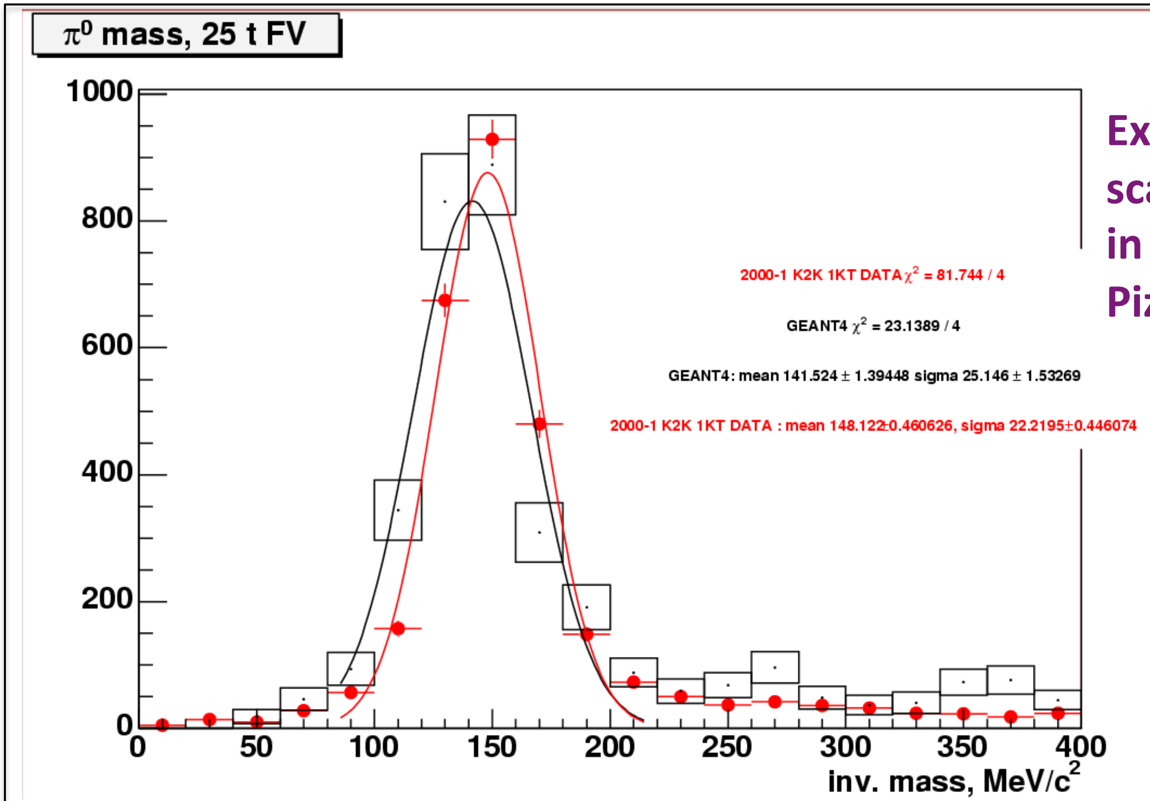
The top volume of the geometry is specified

```
<?xml version="1.0" encoding="UTF-8" ?>

<gdml xmlns:gdml="http://cern.ch/2001/Schemas/GDML"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="schema/gdml.xsd" >

 <define>
  <constant name="const1" value="100" />
  <position name="pos1" unit="mm" x="0" y="0" z="0" />
  <rotation name="rot1" unit="deg" x="const1" y="0" z="0" />
  <scale name="scl1" x="-1" y="+1" z="+1"/>
 </define>

 <materials>
   <element Z="7" formula="N" name="Nitrogen" >
     <atom value="14.01" />
   </element>
   <element Z="8" formula="O" name="Oxygen" >
     <atom value="16" />
   </element>

   <material formula=" " name="Air" >
     <D value="0.00128" />
     <fraction n="0.7" ref="Nitrogen" />
     <fraction n="0.3" ref="Oxygen" />
   </material>
 </materials>

 <solids>
   <box aunit="radian" lunit="mm" name="world" x="10000" y="10000" z="10000" />
   <box aunit="radian" lunit="mm" name="det" x="2000" y="2000" z="2000" />
 </solids>

 <structure>
   <volume name="Detector" >
     <materialref ref="Air" />
     <solidref ref="det" />
   </volume>
   <volume name="World" >
     <materialref ref="Air" />
     <solidref ref="world" />
     <physvol>
       <volumeref ref="Detector" />
       <positionref ref="pos1" />
       <rotationref ref="rot1" />
       <scaleref ref="scl1" />
     </physvol>
   </volume>
 </structure>
```

Goals

- GDML setups for:
  - SK size for validation
  - Strawman Design (from Kadel **DUSEL Note # 73)**
  - Larger mailbox design
- 20% and 40% coverage
- Check scattering / reflection etc
- Get the SK reduction chain working on output

Then we need to do real work..



Example: energy scale before tuning in kton tank using Pizero data and MC.

G4 : 141.5 ± 1.5 MeV/c²

Data (2000-2001) : 148.1 ± 0.5 MeV/c²

Without any tuning of the pe->MeV tables (ptasmo…)

Distribution

- I want this code to be available to everyone interested in large WC detectors and hope SK / SK-Gd and DUSEL and Memphis people will contribute to it.
- This will maximize the tuning and utility of the project. This is will be particularly useful for the postdocs we hire who are working on the existing experiments.
- This is similar to GLG4SIM for liquid scintillator.
- I will make this code available to all via SVN.
- We can imagine/engineer “experiment specific extensions” in the future but I don’t think it is necessary now.
- Should be ready for all in a few weeks.

Conclusions

- Modified 2KM G4 Code for use as the DUSEL WC MC.
- Computation speeds look reasonable.
- The base code is almost ready for distribution.
- The next basic step is geometry layout and validation against existing data.